

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2000-015873

(43)Date of publication of application : 18.01.2000

(51)Int.Cl.

B41J 5/30

G06T 1/00

H04N 1/41

(21)Application number : 10-184440

(71)Applicant : FUJI XEROX CO LTD

(22)Date of filing : 30.06.1998

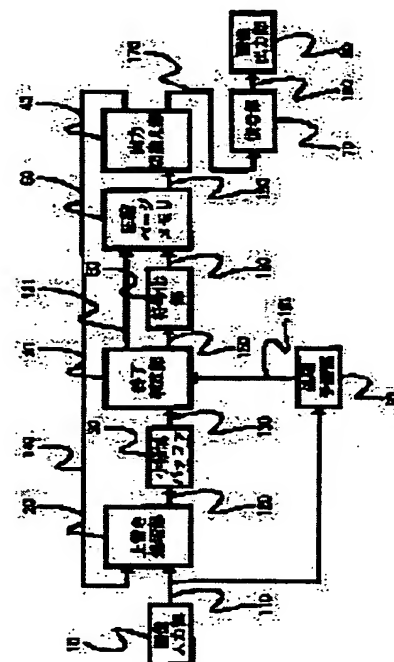
(72)Inventor : YOKOSE TARO
SO KAZUNORI

(54) APPARATUS AND METHOD FOR PROCESSING IMAGE

(57)Abstract:

PROBLEM TO BE SOLVED: To overwrite images inexpensively and at high speed.

SOLUTION: An image input part 10 sequentially inputs partial images of an image. An overwrite process part 20 takes out a corresponding partial image (partial image not coded) from a compression page memory 60, overwrites the input partial image to the partial image and stores in a small area buffer 30. A process estimation part 90 analyzes a plurality of images to be overwritten and estimates whether or not the overwritten partial image is to be further overwritten. When the overwritten partial image is judged to be overwritten, the partial image is coded at a coding part 50 and updated, written in the compression page memory 60. On the other hand, when the image is judged to be not to be overwritten, the partial image is updated, written in the compression page memory 60 without being coded. At a time point when all partial images of all images are completely processed, images completely overwritten are stored in the compression page memory 60 in a state in which the images are coded.



LEGAL STATUS

[Date of request for examination]

17.10.2002

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the
examiner's decision of rejection or application
converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of
rejection][Date of requesting appeal against examiner's decision
of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号
特開2000-15873
(P2000-15873A)

(43) 公開日 平成12年1月18日 (2000.1.18)

(51) Int.Cl. ⁷	識別記号	F I	テームト* (参考)		
B 4 1 J	5/30	B 4 1 J	5/30	Z	2 C 0 8 7
G 0 6 T	1/00	H 0 4 N	1/41	B	5 B 0 5 7
H 0 4 N	1/41	G 0 6 F	15/66	4 5 0	5 C 0 7 8

審査請求 未請求 請求項の数23 O L (全 19 頁)

(21) 出願番号 特願平10-184440

(22) 出願日 平成10年6月30日 (1998.6.30)

(71) 出願人 000005496

富士ゼロックス株式会社
東京都港区赤坂二丁目17番22号

(72) 発明者 横瀬 太郎

神奈川県足柄上郡中井町境430 グリーン
テクなかい 富士ゼロックス株式会社内

(72) 発明者 宋 一憲

神奈川県足柄上郡中井町境430 グリーン
テクなかい 富士ゼロックス株式会社内

(74) 代理人 100086531

弁理士 澤田 俊夫

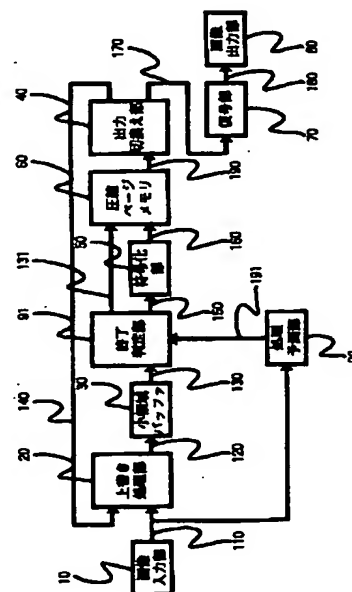
最終頁に続く

(54) 【発明の名称】 画像処理装置および方法

(57) 【要約】

【課題】 画像の上書き処理を安価かつ高速に行なう。

【解決手段】 画像入力部10は画像の部分画像を順次入力する。上書き処理部20は、圧縮ページメモリ60から対応する部分画像（符号化されていない部分画像）を取出して、入力部分画像をこれに上書きして小領域バッファ30にストアする。処理予測部90は、複数の上書き対象画像を解析し、上書きした部分画像がさらに上書きされるかどうかを予測する。上書きされないと予測されたときには、部分画像を符号化部50で符号化して圧縮ページメモリ60に更新書き込みする。他方、上書きされないと予測されたときには、部分画像を符号化することなく圧縮ページメモリ60に更新書き込みする。すべての画像のすべての部分画像の処理が終了した時点では圧縮ページメモリ60には上書き処理が終了した画像が符号化されてストアされている。



【特許請求の範囲】

【請求項1】 画像を入力する画像入力手段と、

前記入力された画像と後述する出力切換え手段から送出される画像に対して部分画像単位の上書き処理を行う上書き処理手段と、

前記上書き処理手段の結果を保持する部分画像格納手段と、

前記部分画像格納手段の部分画像について処理予測を行う処理予測手段と、

前記処理予測手段の結果に従って後述する符号化手段またはデータ格納手段に前記部分画像の出力先を切換える終了判定手段と、

前記終了判定手段から送出される部分画像に所定の手法で画像圧縮符号化を施し符号を得る符号化手段と、

前記終了判定手段から送出される前記部分画像か、または前記符号化手段から送出される前記符号のいずれか1つを蓄積するデータ格納手段と、

前記データ格納手段の出力が前記部分画像ならば前記上書き処理手段へ、前記符号ならば後述する復号手段へ出力先を切換える出力切換え手段と、

前記出力切換え手段から送出される符号に対して前記符号化手段で行われる符号化処理の逆処理にあたる復号処理を施す復号手段と、

前記復号手段で得られる画像を出力する画像出力手段とを有し、

前記処理予測は所定の部分画像にそれ以降上書き処理がされるかどうかの予測を行い、前記終了判定手段では上書き処理が終了した部分画像のみを符号化手段へ送るようにしたことを特徴とする画像処理装置。

【請求項2】 画像を入力する画像入力手段と、

前記入力された画像と後述する出力切換え手段から送出される画像に対して部分画像単位の上書き処理を行う上書き処理手段と、

前記上書き処理手段の結果を保持する部分画像格納手段と、

前記部分画像格納手段の部分画像について処理予測を行う処理予測手段と、

前記処理予測手段の結果に従って後述する符号化手段またはデータ格納手段に前記部分画像の出力先を切換える終了判定手段と、

前記終了判定手段から送出される部分画像に所定の手法で画像圧縮符号化を施し符号を得る符号化手段と、

前記終了判定手段から送出される前記部分画像か、または前記符号化手段から送出される前記符号のいずれか1つを蓄積するデータ格納手段と、

前記データ格納手段の出力が前記部分画像ならば前記上書き処理手段へ、前記符号ならば後述する符号変換手段へ出力先を切換える出力切換え手段と、

前記出力切換え手段から送出される符号に対してさらに符号量を削減するような符号変換を行う符号変換手段

と、

前記符号変換手段で得られる符号を出力する符号出力手段とを有し、

前記処理予測は所定の部分画像にそれ以降上書き処理がされるかどうかの予測を行い、前記終了判定手段では上書き処理が終了した部分画像のみを符号変換手段へ送るようにしたことを特徴とする画像処理装置。

【請求項3】 画像を入力する画像入力手段と、

前記入力された画像と後述する第1のデータ格納手段から送出される画像に対して部分画像単位の上書き処理を行う上書き処理手段と、

前記上書き処理手段の結果を保持する部分画像格納手段と、

前記部分画像格納手段の前記部分画像を蓄積する第1のデータ格納手段と、

前記部分画像格納手段から送出される部分画像に所定の手法で画像圧縮符号化を施し符号を得る符号化手段と、

前記符号を蓄積する第2のデータ格納手段と、

前記第2のデータ格納手段の符号に対して前記符号化手段で行われる符号化処理の逆処理にあたる復号処理を施す復号手段と、

前記復号手段で得られる画像を出力する画像出力手段とを有し、

前記部分画像を画像と符号の形態でそれぞれ第1のデータ格納手段と第2のデータ格納手段に格納することを特徴とする画像処理装置。

【請求項4】 画像を入力する画像入力手段と、

前記入力された画像と後述する第1のデータ格納手段から送出される画像に対して部分画像単位の上書き処理を行う上書き処理手段と、

前記上書き処理手段の結果を保持する部分画像格納手段と、

前記部分画像格納手段の前記部分画像を蓄積する第1のデータ格納手段と、

前記部分画像格納手段の部分画像に所定の手法で画像圧縮符号化を施し符号を得る符号化手段と、

前記符号を蓄積する第2のデータ格納手段と、

前記第2のデータ格納手段の符号に対してさらに符号量を削減するような符号変換を行う符号変換手段と、

前記符号変換手段で得られる符号を出力する符号出力手段とを有し、

前記部分画像は画像と符号の形態でそれぞれ第1のデータ格納手段と第2のデータ格納手段に格納することを特徴とする画像処理装置。

【請求項5】 前記符号化手段および復号手段で行われる符号化および復号処理は前記部分画像を独立の画像として符号化することを特徴とする請求項1ないし4に記載の画像処理装置。

【請求項6】 前記符号化手段および復号手段で行われる符号化および復号処理は前記部分画像を独立に符号化

し、かつ前記符号変換手段で統合しやすいように情報の一部を符号化しないで格納することを特徴とする請求項1ないし5に記載の画像処理装置。

【請求項7】 前記情報の一部とは部分画像の先頭と最後のブロックの直流成分であることを特徴とする請求項6に記載の画像処理装置。

【請求項8】 前記符号化手段および復号手段で行われる符号化および復号処理は前記部分画像を独立に符号化し、かつ前記符号変換手段で統合しやすいように符号の並び順を考慮して複数の独立な符号から構成するようにしたことを特徴とする請求項1ないし7に記載の画像処理装置。

【請求項9】 前記複数の独立な符号とはブロックライン単位に独立であることを特徴とする請求項8に記載の画像処理装置。

【請求項10】 前記複数の独立な符号は各符号の開始位置や符号長がサイド情報として前記部分画像の符号に含まれていることを特徴とする請求項1ないし9に記載の画像処理装置。

【請求項11】 前記符号変換手段で行われる符号変換処理は前記部分画像ごとの符号を一画像の符号のようにつなげることで符号量を削減することを特徴とする請求項2または4ないし10に記載の画像処理装置。

【請求項12】 前記一画像の符号のようにつなげる処理は、前記部分画像ごとの符号間で隣り合う直流成分の差分を算出し符号化する処理であることを特徴とする請求項11に記載の画像処理装置。

【請求項13】 前記一画像の符号のようにつなげる処理は、前記部分画像ごとの符号のあい間に符号化および復号を初期化するような制御信号を加えてつなげる処理であることを特徴とする請求項11に記載の画像処理装置。

【請求項14】 前記制御信号とはリスタートマーカであることを特徴とする請求項13に記載の画像処理装置。

【請求項15】 前記符号化手段および復号手段で行われる符号化および復号処理はJPEGなどの非可逆符号化であることを特徴とする請求項1ないし14に記載の画像処理装置。

【請求項16】 前記処理予測手段で行われる処理予測は簡易な画像生成処理によって行われることを特徴とする請求項1ないし15に記載の画像処理装置。

【請求項17】 前記処理予測は上書き処理が開始される前に前もって行われていることを特徴とする請求項1ないし16に記載の画像処理装置。

【請求項18】 前記処理予測はその前に予測処理を行った画像の上書き処理に並行して行うことを特徴とする請求項1ないし16に記載の画像処理装置。

【請求項19】 前記部分画像は矩形であることを特徴とする請求項1ないし18に記載の画像処理装置。

【請求項20】 前記部分画像の縦および横の画素数は8の倍数であることを特徴とする請求項19に記載の画像処理装置。

【請求項21】 画像を入力し、前記入力された画像と後述する上書きの終了していない画像に対して部分画像単位の上書き処理を行い、前記部分画像がそれ以降上書きされるか否かの処理予測を行い、前記部分画像の上書き処理が終了していれば所定の手法で画像圧縮符号化を行って符号を生成し、前記部分画像または前記符号を蓄積し、前記符号に前記画像圧縮符号化の逆処理である復号処理を行うことを特徴とする画像処理方法。

【請求項22】 上層の画像を下層の画像に順次の上書きする画像処理装置において、上書き処理中の画像をストアする画像記憶手段と、処理対象の画像の各々について、部分画像を順次に入力する画像入力手段と、

上記入力された部分画像を、上記画像記憶手段に記憶されている対応する部分画像に上書きする上書き処理手段と、

上記上書き処理手段で上書きされた部分画像がさらに上書きされるかどうかを予測する予測手段と、

上記予測手段により、上書きされると予測されたときには、上記上書き処理された部分画像を符号化することなく上記画像記憶手段に記憶し、上記予測手段により、上書きされないと予測されたときには、上記上書き処理された部分画像を符号化した後上記画像記憶手段に記憶する制御手段とを有することを特徴とする画像処理装置。

【請求項23】 上層の画像を下層の画像に順次の上書きする画像処理装置において、

画像の各々について、部分画像を順次に入力する画像入力手段と、

上書き処理中の画像を符号化することなく記憶する第1の画像記憶手段と、

上書き処理中の画像を符号化して記憶する第2の画像記憶手段と、

上記入力された部分画像を、上記第1の画像記憶手段に記憶されている対応する部分画像に上書きする上書き処理手段と、

上記上書き処理された部分画像を符号化することなく上記第1の記憶手段に記憶する第1の制御手段と、

上記上書き処理された部分画像を符号化した後上記第2の記憶手段に記憶する第2の制御手段とを有することを特徴とする画像処理装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は画像データの画像処理技術に関するものであり、特に画像の上書き処理を安価かつ高速に実現しようとする技術に関するものである。

【0002】

【従来の技術】まず本発明が対象とする画像の上書き処理について説明する。

【0003】画像の上書き処理とは所定単位の画像を別の画像の上に重ね書きする処理を指す。例えばPDL (Page Description Language、ページ記述言語)の1種であるPS (PostScript、『Postscriptリファレンスマニュアル第2版』(Adobe Systems, アスキー)など参照)で処理されるような、不透明インクによる重ね書きはこれに含まれる。上で定義した所定単位とはPDLでいうラスタ1枚であってもよいし、描画オブジェクト1個であってもよい。

【0004】上書き処理は画像の空間座標に対して一方向の処理では行えない。以下、これについて説明する。一般の画像処理は、例えば画像の左上から右下に向かって一方向の処理で実現できる。例えば拡大、縮小、空間フィルタといった処理がこれに含まれる(『画像解析ハンドブック』(高木、下田、東京大学出版会)など)。

【0005】一方、上書き処理では上書きしようとする画像が出力画像の空間座標に対してランダムに入力され得る。従って入力順に処理する限り、出力画像上では空間座標順に処理することができなくなる。図13は上書き処理の説明図である。図中、210、211は描画オブジェクトである。ここではオブジェクト210の上にオブジェクト211を上書きする処理を例示している。出力画像上でランダムスキャンが必要となること、また上書き処理にページメモリが必要なことは同図から明らかである。

【0006】このように上書き処理は、ある処理途中のデータに加工を加えて処理を進めていく。これをデータフローとして見ると、いわゆるフィードバックループが発生していることがわかる。一般にフィードバックループは処理時間のオーバーヘッドを発生し、制御を複雑化するので、短い方がよい。

【0007】他方、ページメモリの必要性をコスト的見地から見た場合、メモリは少ない方が好ましい。近年画像出力機器の高画質化につれて画像の解像度が高くなっているため、メモリ単価の低下にもかかわらず依然としてメモリ削減は重要な課題である。そこでページメモリを削減する従来技術について説明する。

【0008】[第1の従来例]第1の従来例として特開平5-31974号公報の手法について説明する。第1の従来例の基本的原理は、上書き処理の終了した画像をページメモリに格納する際に画像圧縮符号化を適用することにより、必要なメモリ量を削減することである。

【0009】図14は第1の従来例の構成図である。ただし従来例の開示内容を損ねず、かつ本発明の説明に沿うよう各用語を変更し、また説明に不要な部分は省略した。図中、10は画像入力部、20は上書き処理部、30は小領域バッファ、40は出力切換え部、50は符号

化部、60は圧縮ページメモリ、70は復号部、80は画像出力部、110は入力画像データ、120、130は処理画像データ、140は格納画像データ、150は出力画像データ、160、170は符号データ、180は復号画像データである。

【0010】以下、図14の各部について説明する。画像入力部10は外部から入力画像データ110を入力する。上書き処理部20は予め定められた小領域を処理単位として、格納画像データ140に入力画像データ110を上書き処理し、処理画像データ120として小領域バッファ30へ送出する。小領域バッファ30は処理画像データ120を格納し、処理画像データ130として出力切換え部40へ送出する。出力切換え部40は上書き処理が終了していれば出力画像データ150として符号化部50へ、そうでなければ格納画像データ140として上書き処理部20へ送出する。符号化部50は出力画像データ150に所定の圧縮符号化を施し、符号データ160として圧縮ページメモリ60へ送出する。圧縮ページメモリ60は符号データ160を蓄積し、全ての小領域に対応する符号データが揃ったら符号データ170として復号部70へ送出する。復号部70は符号化部50の逆変換となるような復号処理を行い、復号画像データ180として画像出力部80へ送出する。画像出力部80は復号画像データ180を外部へ出力する。

【0011】以上の構成に基づいて第1の従来例の画像処理手順について説明する。図15は第1の従来例の動作を説明するフローチャートである。

【0012】画像処理手順の具体的な説明に入る前に用語の定義を行う。第1の従来例では画像を部分画像に分割して扱う。これを小領域と呼ぶ。また現在処理している画像を注目画像、同様に処理している小領域を注目小領域と呼ぶことにする。

【0013】以下、図15を用いて第1の従来例の動作を説明する。S10では画像入力部10において入力画像データ110を入力する。S20では入力画像データ110を小領域単位に独立に扱い、現在入力されている部分が注目小領域に相当すればS30へ、そうでなければS50へ進む。S30では小領域バッファ30に格納されている注目小領域を、出力切換え部40を通して、格納画像データ140として読み出す。S40では上書き処理部20において、格納画像データ140に対して入力画像データ110の上書き処理を行う。S50では入力画像データ110が注目小領域に該当しないので、その小領域を読み飛ばす。S60では注目画像の全ての小領域の処理が終了していればS70へ、そうでなければS20へ進む。S70では全ての入力画像の処理が終了していればS80へ、そうでなければS90へ進む。S80では上書きが終了した出力画像データ150に対して、符号化部50で符号化を行い、圧縮ページメモリ60へ格納する。S90では注目画像を次の画像に移

す。S100では全ての小領域に対して上書き処理が終了していればS110へ、そうでなければS120へ進む。S110では復号部70において符号データ170の復号処理を行い、画像出力80へ出力する。S120では注目小領域を次の小領域に移し、注目画像を入力画像データ110の先頭の画像に移す。

【0014】以上の動作において、符号化部50および復号部70は符号データ160のデータ量が処理画像データ150より小さくなるような画像圧縮符号化を行う。第1の従来例ではDCT (Discrete Cosine Transform、離散コサイン変換)を用いた圧縮の例が示されている。

【0015】第1の従来例によれば上書き処理を終了した画像を符号化して格納するため、画像1枚分の符号量に相当するメモリを用意すればよく、メモリの削減が図れる。以下、このように画像を圧縮して1枚蓄積することを目的としたメモリを、圧縮ページメモリと呼ぶ。

【0016】第1の従来例では符号化処理を出力画像の画素順に行う構成としている。このため出力画像に合わせて入力画像を繰り返し入力する必要がある。これは処理時間の増加という問題を招く。

【0017】[第2の従来例] 以上説明した第1の実施例では処理時間が増加するという問題を有していた。そこで処理時間の増加という問題を解決する第2の従来例が提案されている。第2の従来例として特開平5-37789号公報の手法について説明する。第2の従来例の基本的原理は、上書き処理途中の画像に画像圧縮符号化を行い、ページメモリに格納することにより、必要なメモリ量を削減しつつ画像入力を1度ですませる点にある。

【0018】図16は第2の従来例の構成図である。図16において、図14と同様の部分には同一の符号を付して説明を省略する。141は格納画像データ、181は復号画像データである。

【0019】以上の構成に基づいて第2の従来例の画像処理手順について説明する。図17は第2の従来例の動作を説明するフローチャートである。図17において、図15と同様の部分には同一の符号を付して説明を省略する。

【0020】以下、図17を用いて第2の従来例の動作を説明する。S31では圧縮ページメモリ60に格納されている注目小領域の符号データ170を読み出し、復号部70において復号処理を行い、出力切換え部40を通じて格納画像データ141として上書き処理部20へ入力する。S81では注目小領域の上書き処理が終了しているか否かにかかわらず、符号化部50において処理画像データ130に符号化処理を行う。

【0021】第2の従来例によれば第1の従来例同様に圧縮ページメモリによるメモリの削減が図れ、また上書き処理途中の小領域画像も圧縮ページメモリに格納する

ようにしたので同じ画像を繰り返し入力する必要がない。

【0022】一方、処理中のものも含めて全ての小領域画像を符号するので、符号化処理の負荷が重く処理時間のオーバーヘッドが発生する。また符号化に非可逆符号化を用いる場合、同一画像に符号化を繰り返すので、ジェネレーションノイズと呼ばれる誤差の蓄積が行われ、画質の劣化を招く。

【0023】

【発明が解決しようとする課題】従来例の問題点を整理する。図18は従来例による上書き処理のシーケンス例である。ここでは入力画像1に入力画像2を上書き処理するようなシーケンスを考える。

【0024】第1の従来例によれば小領域222にあたる部分を処理するために、小領域220、221、223を読み飛ばす。他の部分でも同様に読み飛ばしが発生するため、画像入力についてのオーバーヘッドが大きいことがわかる。また第2の従来例によれば小領域220に小領域223を上書き処理する前に、小領域220を復号する必要がある。これは図16の格納画像データ141のフィードバックに相当するが、復号動作を含むためにこのフィードバックループが長くなり、処理時間の増加を招く。また前述したように非可逆符号化の場合には画質劣化の問題もある。

【0025】本発明は上述の事情に鑑みてなされたもので、安価かつ高速に上書き処理を実現する画像処理技術を提供することを目的とする。

【0026】

【課題を解決するための手段】請求項1に記載の発明によれば、画像処理装置を、上書きする画像を入力する画像入力手段と、前記入力された画像に対して部分画像単位で上書き処理を行う上書き処理手段と、前記上書き処理手段の結果を保持する部分画像格納手段と、前記部分画像格納手段に格納された部分画像にそれ以降上書き処理がされるか処理の予測を行う処理予測手段と、前記処理予測手段の結果に従って前記画像格納手段に格納された部分画像の出力先を切換える終了判定手段と、前記終了判定手段から送出される部分画像に所定の手法で画像圧縮符号化を施し符号を得る符号化手段と、前記終了判定手段または前記符号化手段から送出される前記部分画像または前記符号のいずれかを蓄積するデータ格納手段と、前記データ格納手段の出力を切換える出力切換え手段と、前記出力切換え手段から送出される符号に対して前記符号化手段で行われる符号化処理の逆処理にあたる復号処理を施す復号手段と、前記復号手段で得られる画像を出力する画像出力手段とから構成し、前記上書き処理手段で行われる上書き処理は前記出力切換え手段から送出される画像の上に前記画像入力手段から入力される画像の上書きを行い、前記終了判定手段では上書き処理が終了した部分画像のみが符号化手段へ送られるように

している。

【0027】この構成においては、入力画像を繰り返し入力する必要がなく画像入力オーバーヘッドを抑えることができ、しかも、上書き処理が終了していないと予測される部分画像については、符号化することなくデータ格納手段に蓄積するので、上書き処理のフィードバック経路の処理負荷も抑制できる。もちろん、上書き処理が終了していると予測される部分画像については符号化してデータ格納手段に蓄積するのでデータ格納手段のメモリ容量を小さく、コスト低減を図れる。

【0028】なお、上書き処理が以降行われるかどうかの予測は正確なものでもよいし、不正確なものでもよい。不正確な場合には、予測ミス発生時に、予測ミスに応じたペナルティが生じるが、予測ヒットにより全体としての画像処理効率率は向上する。予測の性質上、本来書き込みが終了していないのに終了していると判断されるおそれがある場合には、書き込みが終了していると判断されたときにさらに正確な予測を行なうようにしてもよい。このようにすると、処理負荷が若干重くなる反面、予測ミスによるペナルティがなくなる。

【0029】また、請求項2に記載の発明によれば、前記画像処理装置を、上書きする画像を入力する画像入力手段と、前記入力された画像に対して部分画像単位で上書き処理を行う上書き処理手段と、前記上書き処理手段の結果を保持する部分画像格納手段と、前記部分画像格納手段に格納された部分画像にそれ以降上書き処理がされるか処理の予測を行う処理予測手段と、前記処理予測手段の結果に従って前記画像格納手段に格納された部分画像の出力先を切替える終了判定手段と、前記終了判定手段から送出される部分画像に所定の手法で画像圧縮符号化を施し符号を得る符号化手段と、前記終了判定手段または前記符号化手段から送出される前記部分画像または前記符号のいずれかを蓄積するデータ格納手段と、前記データ格納手段の出力を切替える出力切換え手段と、前記出力切換え手段から送出される符号に対してさらに符号量を削減するような符号変換を行う符号変換手段と、前記符号変換手段で得られる符号を出力する符号出力手段とから構成し、前記上書き処理手段で行われる上書き処理は前記出力切換え手段から送出される画像の上に前記画像入力手段から入力される画像の上書きを行い、前記終了判定手段では上書き処理が終了した部分画像のみが符号化手段へ送られるようにしている。

【0030】この構成においても、画像入力オーバーヘッドおよび上書き処理のフィードバック経路の処理負荷の双方を抑制でき、さらにデータ格納手段のメモリ容量を低減できる。

【0031】また、請求項3に記載の発明によれば、画像処理装置を、上書きする画像を入力する画像入力手段と、前記入力された画像に対して部分画像単位で上書き処理を行う上書き処理手段と、前記上書き処理手段の結

果を保持する部分画像格納手段と、前記部分画像格納手段から送出される前記部分画像を蓄積する第1のデータ格納手段と、前記部分画像格納手段から送出される部分画像に所定の手法で画像圧縮符号化を施し符号を得る符号化手段と、前記符号化手段から送出される前記符号のいずれかを蓄積する第2のデータ格納手段と、前記第2のデータ格納手段から送出される符号に対して前記符号化手段で行われる符号化処理の逆処理にあたる復号処理を施す復号手段と、前記復号手段で得られる画像を出力する画像出力手段とから構成し、前記上書き処理手段で行われる上書き処理は前記第1のデータ格納手段から送出される画像の上に前記画像入力手段から入力される画像の上書きを行い、前記部分画像は画像と符号の形態でそれぞれ第1のデータ格納手段と第2のデータ格納手段に格納するようにしている。

【0032】この構成においては、部分画像への上書き処理が終了しているかいかにかかわらず、部分画像を符号化できるので画像処理のスループットを向上させることができる。部分画像への上書きが終了してなくても上書き処理を行わなくてはならない場合でも、第1のデータ格納手段に格納されている、符号化されていない画像を用いることができ、この結果、上書き処理のフィードバックループから符号化・復号の処理を省くことができ、処理負荷が抑制される。

【0033】また、請求項4に記載の発明によれば、画像処理装置を、上書きする画像を入力する画像入力手段と、前記入力された画像に対して部分画像単位で上書き処理を行う上書き処理手段と、前記上書き処理手段の結果を保持する部分画像格納手段と、前記部分画像格納手段から送出される前記部分画像を蓄積する第1のデータ格納手段と、前記部分画像格納手段から送出される部分画像に所定の手法で画像圧縮符号化を施し符号を得る符号化手段と、前記符号化手段から送出される前記符号のいずれかを蓄積する第2のデータ格納手段と、前記第2のデータ格納手段から送出される符号に対してさらに符号量を削減するような符号変換を行う符号変換手段と、前記符号変換手段で得られる符号を出力する符号出力手段とから構成し、前記上書き処理手段で行われる上書き処理は前記第1のデータ格納手段から送出される画像の上に前記画像入力手段から入力される画像の上書きを行い、前記部分画像は画像と符号の形態でそれぞれ第1のデータ格納手段と第2のデータ格納手段に格納するようにしている。

【0034】この構成においても、画像処理のスループットが向上し、しかも、上書き処理のフィードバック経路から符号化・復号の処理を省くことができる。

【0035】また、請求項5に記載の発明によれば、請求項1ないし4に記載の画像処理装置において、前記符号化手段および復号手段で行われる符号化および復号処理は前記部分画像を独立の画像として符号化するように

している。

【0036】また、請求項6に記載の発明によれば、請求項1ないし5に記載の画像処理装置において、前記符号化手段および復号手段で行われる符号化および復号処理は前記部分画像を独立に符号化し、かつ前記符号変換手段で統合しやすいように情報の一部を符号化しないで格納するようにしている。

【0037】また、請求項7に記載の発明によれば、請求項6に記載の画像処理装置において、前記符号化を行わずに格納する情報の一部は、部分画像の先頭および最後のブロックの直流成分としている。

【0038】また、請求項8に記載の発明によれば、請求項1ないし7に記載の画像処理装置において、前記符号化手段および復号手段で行われる符号化および復号処理は前記部分画像を独立に符号化し、かつ前記符号変換手段で統合しやすいように符号の並び順を考慮して複数の独立な符号から構成するようにしている。

【0039】また、請求項9に記載の発明によれば、請求項8に記載の画像処理装置において、前記複数の独立な符号はブロックライン単位に独立であるようにしている。

【0040】また、請求項10に記載の発明によれば、請求項1ないし9に記載の画像処理装置において、前記複数の独立な符号の各々の開始位置や符号長がサイド情報として前記部分画像の符号に含まれるようにしている。

【0041】また、請求項11に記載の発明によれば、請求項2、または請求項4ないし10に記載の画像処理装置において、前記符号変換手段で行われる符号変換処理は前記部分画像ごとの符号を一画像の符号のようにつなげることで符号量を削減するようにしている。

【0042】また、請求項12に記載の発明によれば、請求項11に記載の画像符号化装置において、前記一画像の符号のようにつなげる処理を、前記部分画像ごとの符号間で隣り合う直流成分の差分を算出して符号化する処理としている。

【0043】また、請求項13に記載の発明によれば、請求項11に記載の画像処理装置において、前記符号変換手段で行われる符号変換処理を、前記部分画像ごとの符号のあい間に符号化および復号を初期化するような制御信号を加えてつなげる処理としている。

【0044】また、請求項14に記載の発明によれば、請求項13に記載の画像処理装置において、前記制御信号をリスタートマーカーとしている。

【0045】また、請求項15に記載の発明によれば、請求項1ないし14に記載の画像処理装置において、前記符号化手段および復号手段で行われる符号化および復号処理を、JPEG (Joint Photographic Experts Group) などの非可逆符号化としている。

【0046】また、請求項16に記載の発明によれば、請求項1ないし14に記載の画像処理装置において、前記処理予測手段で行われる処理予測を、所定部分画像がどの入力画像に上書きされるかを調べる処理としている。

【0047】また、請求項17に記載の発明によれば、請求項1ないし15に記載の画像処理装置において、前記処理予測を、上書き処理が開始される前に前もって行うようにしている。

【0048】また、請求項18に記載の発明によれば、請求項1ないし16に記載の画像処理装置において、前記処理予測を、その前に予測処理を行った画像の上書き処理と並行して行うようにしている。

【0049】また、請求項19に記載の発明によれば、請求項1ないし18に記載の画像処理装置において、前記部分画像を矩形としている。

【0050】また、請求項20に記載の発明によれば、請求項19に記載の画像処理装置において、前記部分画像の縦および横の画素数を8の倍数としている。

【0051】さらに、本発明は、特許請求の範囲の他の請求項の構成を採用することもできることはもちろんである。また、本発明は、方法の態様で実現することも可能であり、または少なくともその一部をコンピュータ・ソフトウェアの態様で実現することも可能である。

【0052】

【発明の実施の形態】以下、本発明の実施例について述べる。

【0053】【第1の実施例】本発明の第1の実施例を具体的に説明する前に、本発明の基本的な原理について述べる。従来例で課題にあげたような問題点が発生する理由は、上書き処理のフィードバックループの扱いにあるといえる。

【0054】第1の従来例では、圧縮ページメモリを実現するための符号化および復号処理をフィードバックループから除いているため、注目小領域に限ってみれば高速な処理を実現できる。しかし、この構成を実現するために、処理シーケンスとして大きなフィードバックループを組み込むことになった。これは図15のフローチャートに示した、注目小領域の遷移に伴うループを指す。このため同一画像を何度も入力することになり、逆に処理時間を増大する結果となった。

【0055】また第2の従来例では、処理シーケンスのフィードバックループを避けるため、小領域単位の上書き処理に圧縮ページメモリへの格納を、データフローとして組み込んだ。この結果処理シーケンスは簡略化されたが、フィードバックループに本来必要ない符号化および復号処理が組み込まれたため、ループが長くなり処理時間を増大することになった。

【0056】そこで本発明ではフィードバックが必要なデータ（あるいはそのように予測されるデータ）は符号

化せず、そのまま出力できるデータのみを符号化することを考える。そのため小領域毎に画像を解析し、以降の処理パスの予測を行う。その結果としてフィードバックが必要なデータ、すなわち上書き処理が終了していない小領域は符号化せず、またフィードバックが必要ないデータ、すなわち上書き処理が終了した小領域は符号化して格納する。

【0057】以下、本発明の第1の実施例について具体的に説明する。図1は本発明の第1の実施例を示すブロック図である。図1において、図14または図16と同様の部分には同一の符号を付して説明を省略する。90は処理予測部、91は終了判定部、131は処理画像データ、190は格納データ、191は処理予測データである。

【0058】次に図1の各部について説明する。処理予測部90は注目小領域の上書き処理について、最後の上書きか否かを予測する。処理予測の具体例については後述する。終了判定部91は処理予測部90による予測結果に基づいて、上書き処理が終了しているデータは出力画像データ150として符号化部50へ、そうでないデータは処理画像データ131として圧縮ページメモリ60へ送出する。

【0059】以上の構成に基づいて第1の実施例の動作について説明する。図2は第1の実施例の動作を示すフローチャートである。図中、図15、図17と同様の部分には同一の符号を付して説明を省略する。

【0060】以下、図2を用いて第1の実施例の画像処理手順について説明する。S210では処理画像データ130の上書き処理が終了していればS80へ、そうでなければS81へ進む。S81では処理画像データ131を符号化せず、そのまま圧縮ページメモリ60へ格納する。

【0061】以上の動作の中で、処理予測部90における処理予測について説明する。処理予測は各小領域に残された処理があるかどうかを予測する。このため、ある小領域がどの入力画像に関連しているかを調べるが、これは画素値情報を必要としないため通常のPDLのデコンポーズより軽い処理で行える。特にラスタオブジェクトを貼り込むような処理は相対的に高速に行える。

【0062】図3は処理予測の概念を説明したものである。いま同図(a)を最終的に得たいとする。このときPDLデコンポーズは同図(b)ないし(e)に示すような情報を保持している。すなわち同図(b)は1枚目の入力画像の画素値情報、同図(c)は1枚目の入力画像の有効領域の外形を表す情報である。同図(c)のような外形情報を、以下形状情報と呼ぶ。同様に同図

(d)、同図(e)はそれぞれ2枚目の入力画像の画素値情報、形状情報である。さて同図(a)をPDLデコンポーズで生成する場合、一般には同図(c)および

(e)から同図(f)のような重ねた形状情報を作成し

ておき、その後同図(b)、(d)を参照して最終画像(a)を生成する。一方本発明に必要なものは、上で述べた処理予測の結果だから、同図(f)のような情報で十分であり、後半の処理が省ける。前述したようなラスタオブジェクトを貼り込むような処理の場合などは、この後半の処理負荷が非常に重いので本実施例の効果が高くなる。

【0063】以下に処理予測について詳細に説明する。ここでは例としてPDL処理データの形態であるエッジリスト上での処理をとりあげる。エッジリストとはオブジェクトの識別子とそのオブジェクトがラスタスキャン方向に連続する個数を1つの単位情報とし、この単位情報の羅列で画像全体を表現する形式を指す。特定の小領域の処理予測は、その小領域内にオブジェクト間の切れ目(エッジ)があるかないかで判断することができる。

【0064】この処理の様子を、図4を用いて説明する。同図(a)に示す外形情報のうち特定のラインに注目したのが同図(b)であり、これをエッジリスト表現したのが同図(c)である。いま小領域の幅を8画素とする。同図(b)に含まれる3つの小領域について、同図(c)におけるオブジェクト間のエッジの有無からだけで判定すると、中央の小領域のみ上書きが行われることがわかる。左および右の小領域にもエッジが存在するが、これはオブジェクト間のエッジではないので条件に該当しない。以上は各小領域の最上ラインに対しての処理なので、最終的には同様の処理を小領域中の各ラインについて行う必要がある。

【0065】以上エッジリスト上での処理予測を説明したが、他にもいくつかの手法が考えられる。例えば小領域をn画素×n画素ブロックとした場合、解像度を1/nに落として描画することで小領域単位の上書きの有無を確認することができる。この場合、たまたま小領域と同じ大きさで上書きするケースを上書きありと誤判定する可能性があるが、そのままでも総体的な影響は少ないし、処理の精度をその部分だけ変更すれば誤判定を回避できる。またデコンポーズ処理では一般にエッジリストの手前でディスプレイリストという2次元情報の形態をとるが、ここからエッジリストと同様な処理予測を行ってもよい。こうすれば1次元情報であるエッジリストよりも少ない処理数で処理予測を行うことができる。

【0066】また以上の図4の例では誤判定のない処理予測について説明したが、誤判定を許すかわりに処理負荷を削減するようなバリエーションも考えられる。上述の解像度を落とした描画の例はこの一例で、他にも上書きのありそうな部分だけ局所的に処理予測したり、複雑な形状情報を簡単な図形で近似して処理予測することが考えられる。またエッジリストを複数ラインおきに調べようとして予測してもよい。

【0067】このような場合に起こる誤判定のうち、上

書きがない小領域を上書きありとする誤判定については、処理終了後の圧縮ページメモリ60に処理画像データ131が残ることになる。従って処理終了後にそのようなデータを符号化するステップを追加する必要がある。逆に上書きのある小領域を上書きなしとする誤判定については、上書きしようとする小領域が符号データ160として圧縮ページメモリ60に格納されることになる。このような誤判定は起きないように量子化の方向などで制御することが望ましい。これは例えば上書きなしと判断された小領域のうち近似誤差によって誤判定された可能性のあるものを検出し、それらを上書きありとして再判定する、といった制御を指す。このような制御が不可能な場合は第2の従来例のように局所的に復号する処理が必要になる。ただしこの復号処理は多く発生しないので、処理時間、画質ともに第2の従来例ほどの問題は発生しない。

【0068】さて以上で説明してきた処理予測を事前に、あるいは1ページ前の画像の上書き処理と並行して行い、各小領域に対してどの入力画像が最後の上書き画像になるかを記憶しておけば、あとはリアルタイムで終了判定部91の動作を制御することができる。処理予測の結果のデータフォーマット例を図5に示す。図5に示すように、小領域ごとに最終上書き画像の識別子（#1、#2等）が記録されている。

【0069】ここでいう小領域とは画像全体に対して部分画像であることを指し、小領域の横幅が必ずしも画像の横幅と一致する必要はない。小領域のサイズは装置内のバッファサイズや処理単位によって最適値が変化する。ただし本発明が目的とする小領域別の処理だけと考えた場合には、細かく区切る方が、処理が終了して圧縮できる小領域が多く発生し、本発明の効果がより顕在化する。

【0070】また、入力画像データ110は任意形状での上書きを実現するために画素値情報と形状情報のセットとする。図6は入力画像データ110のデータフォーマット例である。同図(a)はもっとも一般的なフォーマットで、画素値情報と形状情報が1つのデータストリームとなった場合である。もちろん画素値情報と形状情報の順序が逆でもよい。同図(b)は画素単位に画素値情報と形状情報を持った場合である。この場合形状情報は上書きするかしないかを示す。同図(c)は形状情報を小領域単位に分割し、小領域内の画素が全て上書きするかまたは全て上書きしない場合、形状有無を”なし”として小領域の形状情報を1つにまとめてしまう場合である。このようにすれば冗長な形状情報が不要となる。もちろん小領域ごとに画素値情報と形状情報がセットになったようなデータ順でもよい。同図(d)は画像入力部で各小領域について、処理中の画像より後に上書きがあるかどうか知ることができる場合のフォーマットである。この情報を形状情報に含めた以外は、ほぼ同図

(a)と同じ構成となっている。このような場合、図1の処理予測データ191は入力画像データ110から直接生成することが可能となるので、処理予測部90は簡略化できる。

【0071】また、本実施例に用いる符号化手法は可逆でも非可逆でも構わない。ただし本実施例によれば画像の全ての小領域がただ1度だけしか符号化されないの、非可逆符号化を応用してもジェネレーションノイズを発生しない。非可逆符号化の代表例としては、例えば多値画像圧縮符号化の国際標準でもあるJPEG（『カラー静止画像の国際標準符号化方式—JPEGアルゴリズム—』（遠藤、インターフェース、Dec. 1991）など）のベースライン手法（以下、JPEGと呼ぶ）があげられる。

【0072】本実施例では各小領域を独立な画像として符号化する。前述のJPEGアルゴリズムを始め一般的な画像符号化技術は画像を連続した情報として扱うので、符号になってしまうと分離が難しい。一方、本実施例では小領域ごとに処理が終了したか否かで異なる扱いをするので、始めから異なる画像として符号化しておくのが好ましいためである。

【0073】本実施例の効果を確認するために理論的な処理時間を比較する。まず変数について、1画像を小領域画像に分解したときの個数をS、全ての入力画像に含まれる小領域画像の総数をAとする。入力されるA個の小領域画像に対して上書き処理をしてS個の小領域画像を出力するので、必ず $S \leq A$ である。また上書き処理と符号化復号処理の処理時間をそれぞれ T_o 、 T_c とする。上書き処理は負荷が軽く並列化も容易なために一般には $T_o < T_c$ の関係がある。

【0074】第1の従来例は画像の上書き時間が $S \times A \times T_o$ 、また符号化時間が $S \times T_c$ となるが、両処理を並列動作させるとして総処理時間は $\max(S \times A \times T_o, S \times T_c)$ 。次に第2の従来例の上書き時間は復号がボトルネックとなりうるので $\max(A \times T_o, A \times T_c)$ 、符号化時間は考えなくてよい。これに対して本発明では上書き処理と符号化処理が並行して行われ、遅い方がボトルネックになるので $\max(A \times T_o, S \times T_c)$ となる。

【0075】図7は上書きする入力画像の枚数を横軸として各構成の処理時間を比較したものである。ただし3500万画素の画像を1MByteの小領域に分割したと想定し $S = 35$ 、また上書き処理は4画素同時に処理できると想定して $4T_o = T_c$ と仮定した。縦軸は単位時間当たりの処理枚数であり、大きい方が高速処理であることを示す。同図より本発明の効果は明らかである。

【0076】本実施例によれば上書き処理に対するフィードバックループに符号化処理や復号処理といった本来必要ない処理が含まれないので、処理時間が短縮できる。一方上書きを終了した画像は符号化して圧縮ページ

メモリに蓄積するので、必要なメモリ量を削減することができる。また圧縮効率の高い非可逆符号化を適用した場合も、符号化が一度しか行われないので画質劣化が少ない。

【0077】〔第2の実施例〕本発明の第2の実施例として復号処理を含めない画像処理装置について述べる。画像圧縮符号化は第1の実施例で用いた一時的蓄積の用途以外にも、データベースのような長期的蓄積や通信に用いることがある。そこでそのような後処理を想定し、圧縮した符号をそのまま出力とする実施例について述べる。

【0078】まず第2の実施例の基本的な考え方について説明する。第1の実施例で述べたように図1の圧縮ページメモリ60には、小領域ごとに独立な画像として符号化した結果が蓄積される。第1の実施例のような一時的な蓄積にはこれで十分だが、上述の用途を目的とした第2の実施例では符号量が重要な要素になる。符号量を小さくするためには、一般的に画像をなるべく連続したものとして扱った方が好ましい。前出のJPEGの例でいえば、ブロック間の直流成分の差分を符号化するのだが、その処理がまさにこの例にあたる。また符号データのハンドリングという点から考えても、小領域ごとの複数の符号を1つにまとめた方が好ましい。

【0079】そこで第2の実施例では蓄積された符号を出力する前に連続的な符号への変換を行うことを特徴とする。以下、第2の実施例について具体的に説明する。図8は本発明の第2の実施例のブロック図である。図中、図1と同様の部分は同一の符号を付して説明を省略する。71は符号変換部、81は符号出力部、182は出力符号データである。

【0080】図8の各部について説明する。符号変換部71は符号データ170に所定の符号変換を施し、出力符号データ182として符号出力部81へ送出する。符号出力部81は出力符号データ182を外部へ出力する。

【0081】動作については、ほとんど、第1の実施例の説明から容易に類推が可能なので、その範囲で省略する。

【0082】ここでは、符号変換部71における符号変換処理について説明する。図9は本実施例の符号変換処理を説明する説明図である。同図(a)はJPEGの基本的な符号フォーマットを簡単に説明したものである。JPEGの符号は直流成分の差分値の符号と交流成分値の符号で構成され、その前後にヘッダと終了情報(EOI(End Of Image)とよばれる)が配置される。符号データ170が小領域ごとに同図(a)のフォーマットをとっていた場合、出力符号データ182は例えば同図(b)のように構成できる。図中、RST

(ReSTart)はJPEGのマーカーと呼ばれる制御信号の1つで、符号のビット位置と直流成分の差分値

を初期化する役割がある。従って小領域ごとの符号をRSTでつなげば、別々に生成した符号を1枚の画像として復号することができる。以上は出力符号データ182を工夫した場合だが、逆に符号データ170をつなぎやすく構成してもよい。同図(c)はその一例である。同図(c)のフォーマットは後続の符号に含まれる先頭ブロックと最終ブロックの直流成分値を、符号化せずにそのままデータとしている点が特徴である。符号データ170がこの形式であれば、前後の差分値を算出して同図(a)の形式の出力符号データ182が簡単に生成できる。

【0083】以上は小領域が画像全体と同じスキャン順で構成されている場合の議論である。スキャン順について図10を用いて説明する。JPEGの符号化は同図(a)に示すスキャン順で行われる。これに対して同図(b)の小領域1のように小領域を分割すれば、小領域と画像全体を同じスキャン順で処理でき、上述の議論がそのままあてはまる。これに対して同図(b)の小領域2のように小領域を分割すると、スキャン順が変わるので符号データ170をそのままつなぐことができない。図9(d)はそのような場合の符号フォーマット例である。各ブロックライン単位で符号が独立になるように構成し、各符号は同図(c)に従う。ただしブロックラインとは縦がブロックサイズで横が画像サイズであるような、横長の領域を指す。各ブロックラインを独立にすることによって、スキャン順の変換を容易にできる。この場合各ブロックラインの符号の先頭位置や符号長などの構成情報をヘッダに含めてもよい。そのようなサイドデータの例を図10(c)に示す。

【0084】第2の実施例の効果を説明するために、符号データ170と出力符号データ182の符号量を比較する。符号データ170は図9(a)を想定し、画像全体をまとめて符号化する場合に比較して、小領域ごとに300Byteのオーバーヘッドが入ると仮定する。また画像全体をまとめて符号化したときの圧縮率を20、画像サイズを3500万画素と仮定する。図11はこの仮定の下で横軸に小領域のサイズをとった場合の符号データ170と出力符号データ182の符号量比較である。同図から特に小領域のサイズが小さい場合に、本実施例の効果は明らかである。

【0085】なお、以上JPEGを例にとって説明したが、同様に他の画像符号化手法にも本実施例を適用することが可能である。

【0086】本実施例によれば一時的に蓄積した符号を効率のよい符号に変換してから出力するので、より圧縮効率の良い小さい符号量の符号を得ることができる。もちろん、圧縮ページメモリ60からの符号データを直接出力してもよい。

【0087】〔第3の実施例〕本発明の第3の実施例として本発明において処理予測を含めない画像処理装置に

10

20

30

40

50

ついて述べる。第1の実施例で説明したように本発明における処理予測は重い負荷の処理ではないが、実装上の都合などで構成をよりシンプルにしたい場合がある。そのような場合に対応する実施例について述べる。

【0088】まず第3の実施例の基本的な考え方について説明する。処理予測を省略するので符号化した画像に上書き処理が発生する可能性がある。これを復号し始めると第2の従来例のように処理速度および画質が問題になるので、復号処理が不要になるように生の画像を格納する手段を設ける。なお、この場合、ページメモリが必要となるのでコスト的には増加する。

【0089】以下、第3の実施例について具体的に説明する。図12は本発明の第3の実施例のブロック図である。図中、図1または図8と同様の部分は同一の符号を付して説明を省略する。61はページメモリである。

【0090】図12の各部について説明する。ページメモリ61は小領域バッファ30から送られる処理画像データ130を格納し、また格納画像データ140として上書き処理部20へ送出する。

【0091】動作については第1の実施例または第2の実施例から容易に類推可能なので、省略する。

【0092】以上の中で、処理画像データ130は全ての小領域についてページメモリ61と圧縮ページメモリ60に2重に保持する。それ以降に入力された画像に対して上書き処理が発生すればページメモリ61から該当する小領域読み出す。圧縮ページメモリ60に格納したデータは全ての処理が終わった時点で出力する。従ってどの小領域も符号化処理は1度しかかからない。

【0093】本実施例ではページメモリ61を備えるため、第1の実施例や第2の従来例よりコストが高くなる。しかし第1の実施例と処理速度、画質ともに変わらないので、第1の実施例や、第2の実施例同様、処理速度および画質面での効果がある。

【0094】また圧縮ページメモリを使わない場合、例えば第2の実施例の符号化部50、復号部70を省略した構成と比較すると、本実施例の圧縮ページメモリのコストが無視できるのでコスト的には同等である。しかし本実施例では圧縮ページメモリ60の出力中に、次のページの入力を行えるのに対し、上述の例ではページメモリをダブルバッファで持たねばならず、処理時間またはコストが2倍になる。

【0095】本実施例によれば画像を圧縮、非圧縮の2重に格納するようにしたので、予測処理を省いたシンプルな構成をとることができる。

【0096】なお、この実施例の復号部70および画像出力部80に代えて図8に示すような符号変換部71や符号出力部81を設けてもよい。もちろん、圧縮ページメモリ60からの符号データを直接出力するようにしてもよい。

【0097】

【発明の効果】以上の説明から明らかなように、本発明によれば安価でかつ高速な上書き処理装置を実現することができる。また非可逆符号化を適用した場合には高画質な処理を実現できる。さらに圧縮効率の高い符号として出力することができる。また処理予測を省いた構成も可能である。

【図面の簡単な説明】

【図1】 本発明の画像処理装置の第1の実施例を示す構成図である。

【図2】 本発明の画像処理装置の第1の実施例における上書き処理の動作の一例を示すフローチャートである。

【図3】 本発明の画像処理装置の第1の実施例における処理予測の説明図である。

【図4】 本発明の画像処理装置の第1の実施例における処理予測の説明図である。

【図5】 本発明の画像処理装置の第1の実施例における処理予測の結果のデータフォーマット例である。

【図6】 本発明の画像処理装置の第1の実施例における入力画像データ110のデータフォーマット例である。

【図7】 本発明の画像処理装置の第1の実施例の効果の説明図である。

【図8】 本発明の画像処理装置の第2の実施例を示す構成図である。

【図9】 本発明の画像処理装置の第2の実施例における符号変換処理を説明する説明図である。

【図10】 本発明の画像処理装置の第2の実施例におけるスキャン順の説明図である。

【図11】 本発明の画像処理装置の第2の実施例の効果の説明図である。

【図12】 本発明の画像処理装置の第3の実施例を示す構成図である。

【図13】 上書き処理の概念を説明する説明図である。

【図14】 第1の従来例の画像処理装置を示す構成図である。

【図15】 第1の従来例の画像処理装置の動作の一例を示すフローチャートである。

【図16】 第2の従来例の画像処理装置を示す構成図である。

【図17】 第2の従来例の画像処理装置の動作の一例を示すフローチャートである。

【図18】 第1および第2の従来例の処理シーケンスを示す説明図である。

【符号の説明】

10 画像入力部

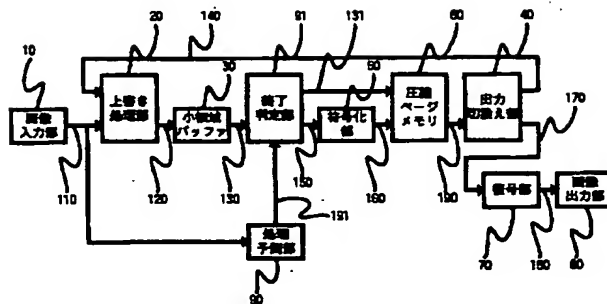
20 上書き処理部

30 小領域バッファ

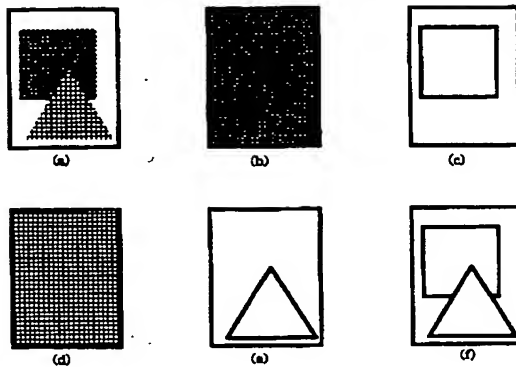
40 出力切換え部

50 符号化部
60 圧縮ページメモリ
61 ページメモリ
70 復号部
71 符号変換部
80 画像出力部
81 符号出力部
90 処理予測部
91 終了判定部
110 入力画像データ
120 処理画像データ
130 処理画像データ

【図1】

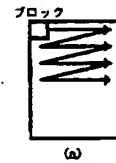


【図3】

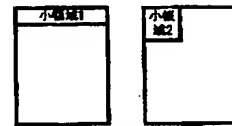


131 処理画像データ
140 格納画像データ
141 格納画像データ
150 出力画像データ
160 符号データ
170 符号データ
180 復号画像データ
181 復号画像データ
182 出力符号データ
190 格納データ
191 処理予測データ

【図10】



(a)

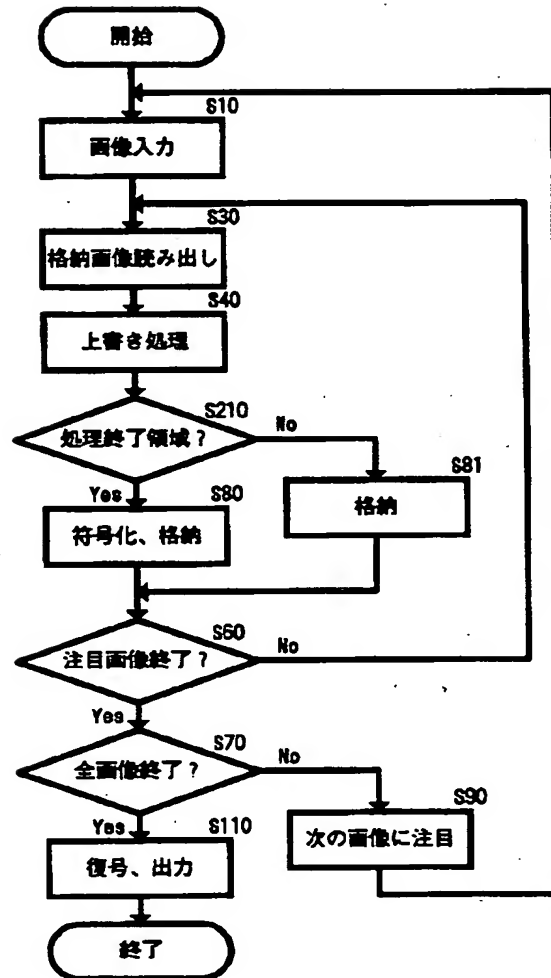


(b)

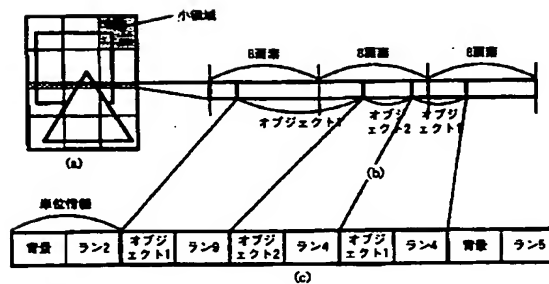
ブロックライン	先頭からのオフセット	符号長
1		
2		
...		
N		

(c)

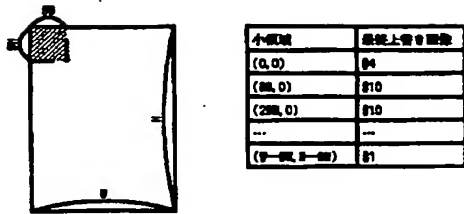
【図2】



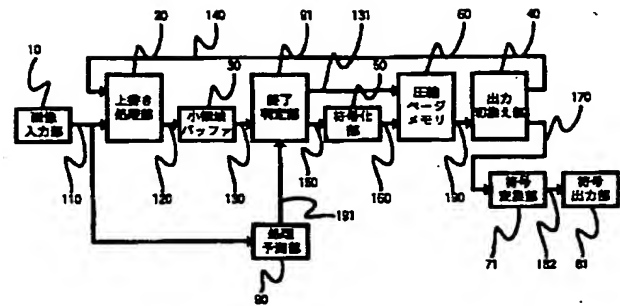
【図4】



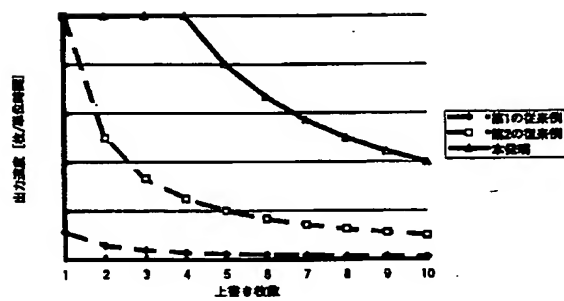
【図5】



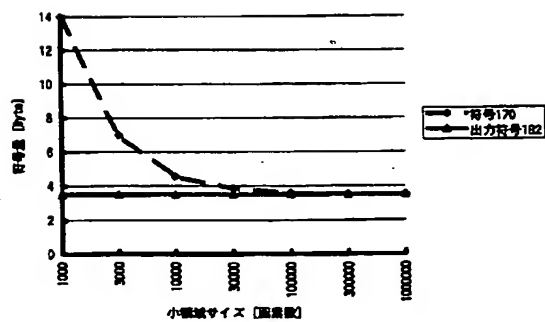
【図8】



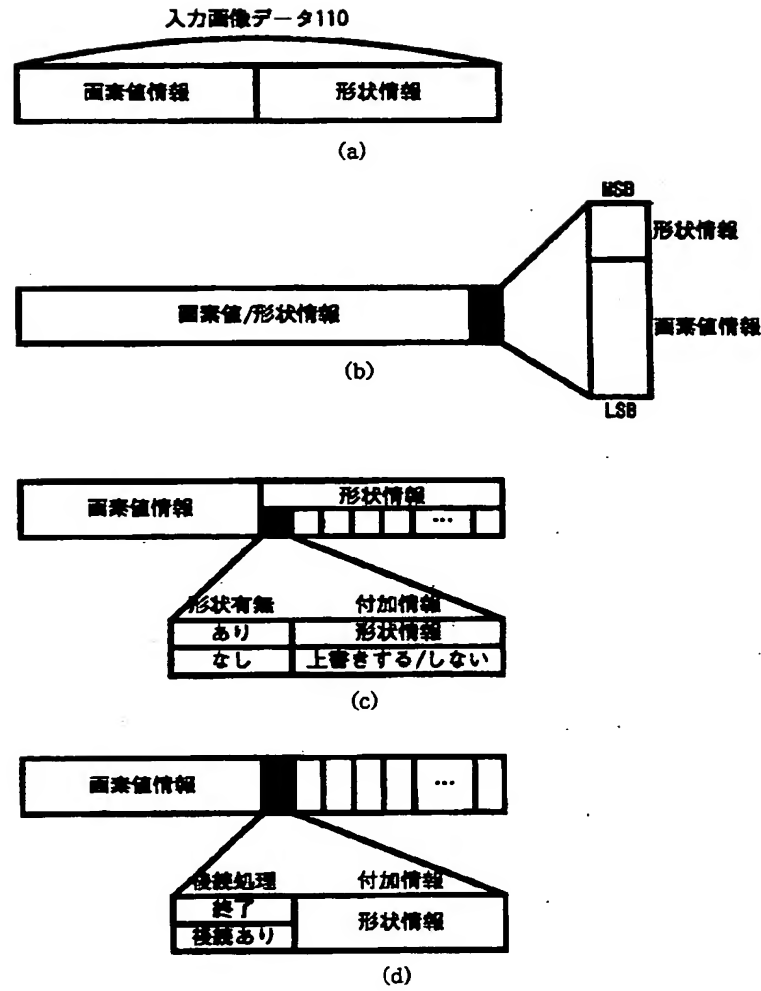
【図7】



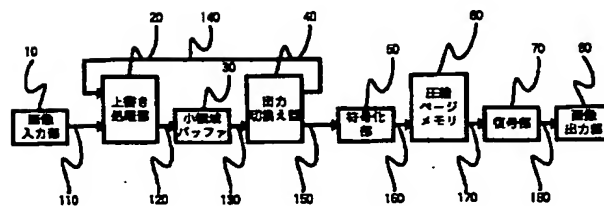
【図11】



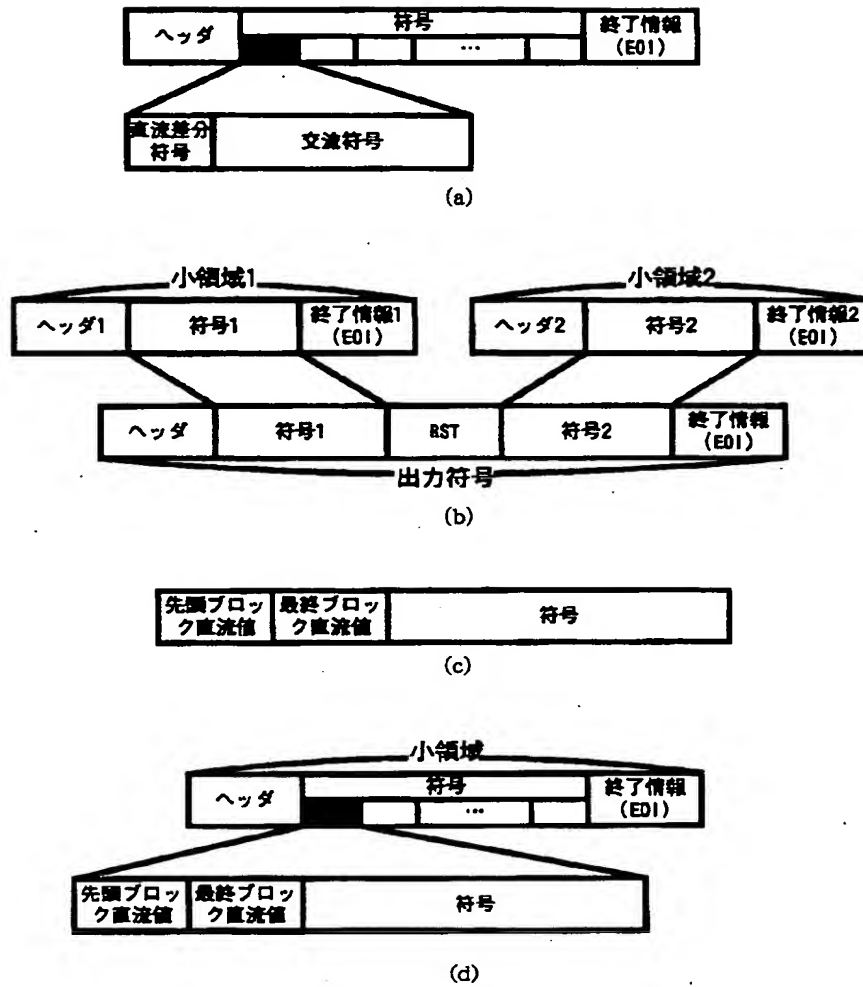
【図6】



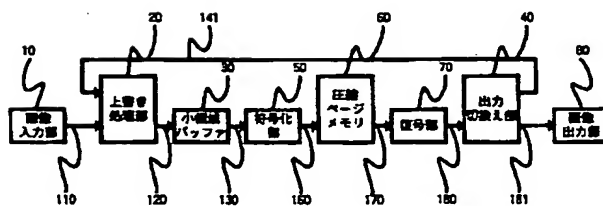
【図14】



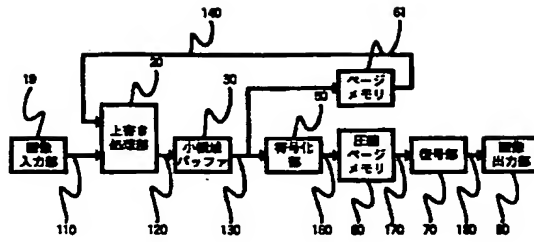
【図9】



【図16】



【図12】



【図18】

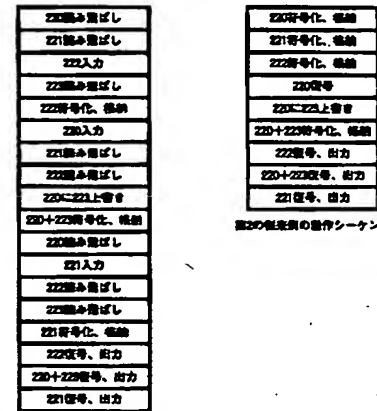
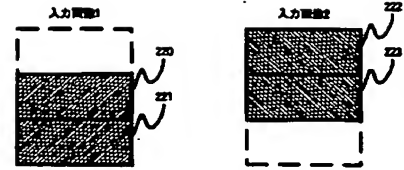
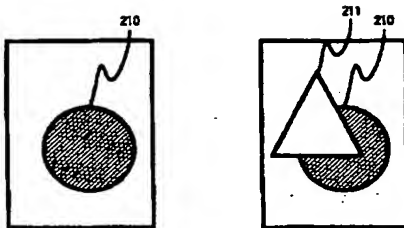


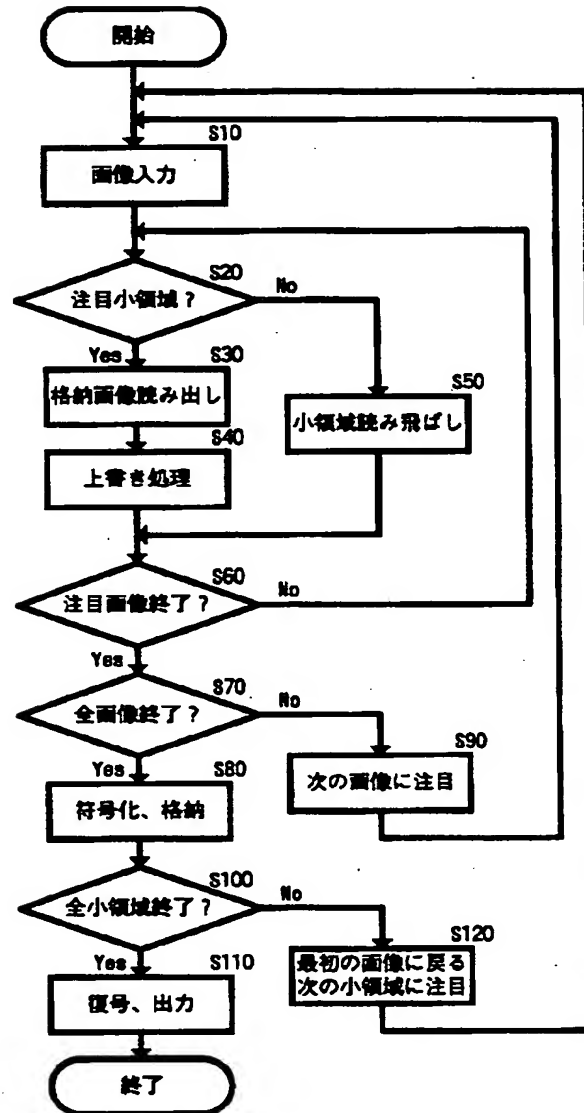
図18の処理例の動作シーケンス

図17の処理例の動作シーケンス

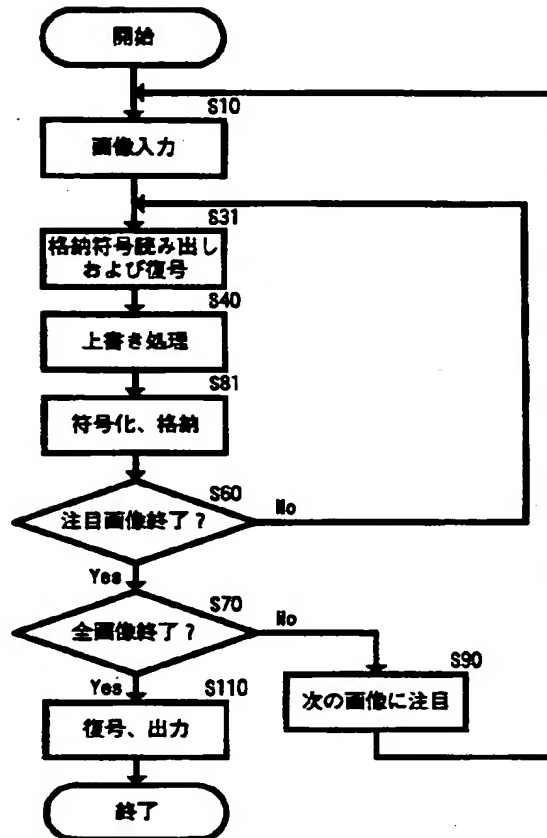
【図13】



【図15】



【図17】



フロントページの続き

Fターム(参考) 2C087 BA03 BC01 BD05 BD12 BD40
5B057 CA16 CB18 CE08 CG07 CH11
CH18
5C078 BA32 CA01 CA12 CA14 CA27
CA31 DA00 DA01 DA02